OBJECT ORIENTED PROGRAMMING WITH JAVA 8– LAB 6

Q1. Create an abstract class Vehicle with one abstract function color(). Create two sub classes Car and Bus from the Vehicle class. Invoke the function through the instance of Car and Bus. Also use the abstract class reference that invokes that function in main.

Ans=

Code-

```
abstract class Vehicle
{
        abstract void color();
}

class Car extends Vehicle
{
        void color()
        {
         System.out.println("This car has a Red color");
        }
}

class Bus extends Vehicle
{
        void color()
        {
         System.out.println("This bus has a White color");
        }
}

class VehicleMain
{

        public static void main(String[] args)
        {
         Car c = new Car();
         Bus b = new Bus();

         c.color();
         b.color();

         Vehicle v = c;
         v.color();

         v = b;
         v.color();
        }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac VehicleMain.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>java VehicleMain
This car has a Red color
This bus has a White color
This car has a Red color
This bus has a White color

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>
```

Q2. Create an abstract class Animals with two abstract methods eat() and makeNoise(). Now create a class Cats where makeNoise() method is overridden which prints "Cats meow" and a class Dogs where makeNoise() method is overridden which prints "Dogs bark", both inheriting the class Animals. Now create an object for each of the subclasses and call their respective methods.

Ans=

Code-

Part1,

```
abstract class Animals
{
        abstract void eat();

        abstract void makeNoise();
}

class Cats extends Animals
{
        void eat()
        {
         System.out.println("Cats eat mice");
        }

        void makeNoise()
        {
         System.out.println("Cats meow");
        }
}

class Dogs extends Animals
{
        void eat()
        {
         System.out.println("Dogs eat meat");
        }

        void makeNoise()
        {
         System.out.println("Dogs bark");
        }
}
```

Part2,

```
class AnimalsMain
{
        public static void main(String[] args)
        {
          Cats c = new Cats();
          Dogs d = new Dogs();

          c.eat();
          d.eat();

          c.makeNoise();
          d.makeNoise();
        }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac AnimalsMain.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>java AnimalsMain
Cats eat mice
Dogs eat meat
Cats meow
Dogs bark

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>
```

Q3. Write an interface called Result with a method checkStatus(int mark) that returns a boolean value. The checkStatus() method should return true if the mark is greater than or equal to 50 else false. Write another interface called Classify with a method checkGrade(int average) which returns a string. The checkGrade() method must return “First class” when the parameter average is 60 or more, “ Second class” when average is 50 or more but below 60, “Third class” when average is less than 50. Write a class called Exam which implements both Result and Classify. Now create an object for Exam class and call their respective methods.

Ans=

Code-

Part1,

```
interface Result
{
        Boolean checkStatus(int mark);
}
interface Classify
{
        String checkGrade(int average);
}

class Exam implements Result
{
        public Boolean checkStatus(int mark)
        {
         return mark >= 50;
        }

        public String checkGrade(int average)
        {
        if (average >= 60)
                {
                 return "First class";
                }
                 else if (average >= 50)
                {
                 return "Second class";
                }
                 else
                {
                 return "Third class";
                }
        }
}
```

Part2,

```
class ExamMain
{
        public static void main(String[] args)
        {
         Exam e = new Exam();

         int mark1= 48;
         boolean status1 = e.checkStatus(mark1);
         System.out.println("The mark is: " + mark1 + "and the Status is: " + status1);

         int mark2= 77;
         boolean status2 = e.checkStatus(mark2);
         System.out.println("The mark is: " + mark2 + "and the Status is: " + status2);

         int average1 = 75;
         String grade1 = e.checkGrade(average1);
         System.out.println(" The Grade is : " + grade1);

         int average2 = 55;
         String grade2 = e.checkGrade(average2);
         System.out.println(" The Grade is : " + grade2);

         int average3 = 45;
         String grade3 = e.checkGrade(average3);
         System.out.println(" The Grade is : " + grade3);
         }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac ExamMain.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>java ExamMain
The mark is: 48and the Status is: false
The mark is: 77and the Status is: true
 The Grade is : First class
 The Grade is : Second class
 The Grade is : Third class

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>
```

Q4. Create one interface named Sample which contains a variable named x and a method meth1(). Implement the interface in a class and implement the method to print the value of the variable. Invoke the method using interface reference.

Ans=

Code-

```
interface Sample
{
        int x = 10;
        void meth1();
}

class SampleImpl implements Sample
{
         public void meth1()
        {
         System.out.println("Value of x: " + x);
        }
}

class SampleMain
{
         public static void main(String[] args)
        {
         SampleImpl s = new SampleImpl();
         s.meth1();
        }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac SampleMain.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>java SampleMain
Value of x: 10
```

Q5. Create an interface, with one abstract method to find square of a number and a default method to find cube of a number. Write the implementing class and invoke the two methods.

Ans=

Code-

```
interface Cube
{
        int square(int num);

        int cube(int num);
}

class Calculation implements Cube
{
        public int square(int num)
        {
         return num * num;
        }

        public int cube(int num)
        {
         return num * num * num;
        }
}

public class CubeMain
{
        public static void main(String[] args)
        {
         Calculation c = new Calculation();
         int n = 7;

         int squareResult = c.square(n);
         System.out.println("Square of " + n + " is: " + squareResult);

         int cubeResult = c.cube(n);
         System.out.println("Cube of " + n + " is: " + cubeResult);
        }
}
```

Execution-

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac CubeMain.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>java CubeMain
Square of 7 is: 49
Cube of 7 is: 343

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>
```

Q6. Create a class named A with following data members:

protected int a

private int b

int c

public int d

Create a no argument constructor which initializes the variables with values a=10, b=20, c=30 and d=40. Create a subclass named B in another package with main function which creates an object of B and print the values of all variables.

Ans= Step 1,

First, we have to create a super class A package called my package. This super class A will contain one protected integer a, one private integer b, one default integer c & one public integer d.

```
package mypackage;

public class A
{
        protected int a;
        private int b;
        int c;
        public int d;

        public A()
        {
        this.a = 10;
        this.b = 20;
        this.c = 30;
        this.d = 40;
        }
}
```

Then we have compiled the super class from the same package named as mypackage from the directory in my command line client.

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer\mypackage>javac A.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer\mypackage>
```

Step 2,

Then we have created a subclass in the main folder & tried to import the values from the super class A .

```
import mypackage.A;

public class B extends A
{
        public static void main(String[] args)
        {
         B b = new B();

         System.out.println("Value of a: " + b.a);
         System.out.println("Value of b: " + b.b);
         System.out.println("Value of c: " + b.c);
         System.out.println("Value of d: " + b.d);
        }
}
```

Then we tried to compile the code from the main directory but it shows error,

Because Super class contains:-

1)Private access specifier integer type variable which can only used within

the same class & not outside the class, Which is int b.

2) Default access specifier integer type variable which can only accessed in same package, which is int c.

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac B.java
B.java:10: error: b has private access in A
        System.out.println("Value of b: " + b.b);
                                             ^
B.java:11: error: c is not public in A; cannot be accessed from outside package
        System.out.println("Value of c: " + b.c);
                                             ^
2 errors

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>
```

Step3,

We have commented the two line which were giving errors to properly compile the code.

```
import mypackage.A;

public class B extends A
{
        public static void main(String[] args)
        {
         B b = new B();

         System.out.println("Value of a: " + b.a);
         //System.out.println("Value of b: " + b.b);
         //System.out.println("Value of c: " + b.c);
         System.out.println("Value of d: " + b.d);
        }
}
```

Then we successfully compiled the code and does not show any error because:-

1) Private access specifier integer type variable a can be Inheritances from super class to sub class.

2) Public access specifier integer type variable d can be accessed from anywhere.

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>javac B.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>java B
Value of a: 10
Value of d: 40

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer>
```

Q7. Write a Java class called Sumn which contains a method that finds the sum of all numbers from 1 upto the given number n. This class is in the directory with path d:\yourname\pack. Write a main program to access this package and print the sum.

Ans=

Step 1,

We have created a Sumn class in the package called Pratik with public specifiers.

```
package Pratik;

public class Sumn
        {
        public int calculateSum(int n)
        {
        int sum = 0;
        for (int i = 1; i <= n; i++)
                {
                 sum += i;
                }
        return sum;
        }
}
```

Then we successfully compiled the Super class in the same package repository

```
C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer\Pratik>javac Sumn.java

C:\Users\p7pha\OneDrive\Desktop\Cdac DBDA\JAVA\Day 6 & Lab6\Answer\Pratik>
```

Step 2,

We created a SumnMain class in main repository & tried to import the values from another Sumn class.

```
import Pratik.Sumn;

public class SumnMain
{
        public static void main(String[] args)
        {
        int n = 10;

        Sumn s = new Sumn();
        int sum = s.calculateSum(n);

        System.out.println("Sum of numbers from 1 to " + n + " is: " + sum);
        }
}
```

At last we successfully compiled the SumnMain class & get the output after execution.